



**UNIVERSIDAD NACIONAL**  
**SEDE REGIONAL CHOROTEGA**

CURSO: Ingeniería en sistemas 3  
PROFESOR: Raymond Pérez Meza  
SEMESTRE: I CICLO

**TEMA:**

Plan de pruebas para el aseguramiento del Software

**Presentado por:**

Nombre			Carné	Rol	Porcentaje de participación
1° Apellido	2° Apellido	Nombre			
Espinoza	Herrera	Abigail	504300604	Front-End	25%
Leal	Villafuerte	Ariana	504280280	Front-End	25%
López	Salas	Yustin	504390350	Back-End	25%
Chaves	Aguirre	Daniela	504000156	Back-End	25%

Para cuestiones de un total de 100% en participación.

# ÍNDICE

<b>Introducción</b>	<b>3</b>
<b>Objetivos</b>	<b>3</b>
<b>Tareas</b>	<b>4</b>
<b>Elementos</b>	<b>4</b>
<b>Módulos sujetos a pruebas:</b>	<b>5</b>
<b>Características a probar</b>	<b>6</b>
<b>Definición de pruebas.</b>	<b>8</b>
<b>Pruebas de rendimiento y estrés</b>	<b>8</b>
<b>Pruebas de aceptación del usuario</b>	<b>10</b>
Pruebas de aceptación del usuario UAT	11
Pruebas de aceptación operativa OAT	11
Pruebas de aceptación de la normativa	11
Pruebas de aceptación Alfa Y Beta	11
<b>Principales entregables</b>	<b>14</b>
<b>Herramientas</b>	<b>15</b>
<b>Bibliografía</b>	<b>16</b>

# Introducción

El propósito de este documento es presentar los resultados de un estudio enfocado en la realización de pruebas para la plataforma SIAVEM, con el objetivo de definir los tipos de pruebas que deben aplicarse para verificar que los niveles de aceptación del sistema hayan sido correctamente realizados. La verificación de estos niveles incluye el diseño funcional, técnico, los requerimientos, la codificación y la especificación de componentes, lo que es esencial para garantizar que la plataforma funcione de manera efectiva y eficiente.

En este estudio se aplicaron diversas pruebas para validar los requisitos y asegurar que la plataforma entregada a la Municipalidad de Santa Cruz cumpla con los estándares de calidad requeridos. La evaluación de los resultados se realizó mediante un proceso riguroso y metódico que permitió detectar posibles fallos y corregirlos antes de su implementación.

Los hallazgos de este estudio pueden ser útiles para futuros proyectos de desarrollo de plataformas similares, al proporcionar información valiosa sobre los tipos de pruebas que deben aplicarse para verificar que los niveles de aceptación del sistema hayan sido correctamente realizados.

## Objetivos

- Identificar los puntos críticos del sistema SIAVEM, incluyendo las áreas de mayor riesgo, y definir pruebas específicas para abordar estos riesgos.

- Realizar un seguimiento continuo del proceso de pruebas, identificando y resolviendo problemas a medida que surjan y actualizando el plan de pruebas según sea necesario.
- Documentar y presentar los resultados de las pruebas de manera clara y concisa para informar a los interesados sobre el desempeño del sistema SIAVEM y asegurar que cumpla con los requisitos de la Municipalidad de Santa Cruz.

## Tareas

- Cronograma del proyecto.
- Requerimientos del proyecto
- Funcionalidades del proyecto
- Estrategias de pruebas.
- Definir los criterios de inicio, aceptación y suspensión de pruebas.
- Identificar los entornos requeridos .
- Metodología de desarrollo del proyecto.

## Elementos

Software Sujeto a Pruebas

Empresa / Organización	Municipalidad de Santa Cruz
Proyecto	SIAVEM (Sistema Integrado de Administración de Vehículos Municipales)
Fecha de entrega	
Líder de proyecto	Daniela Chaves Aguirre
Integrantes	Ariana Leal Villafuerte Yustin Lopez Salas Abigail Espinoza Herrera

# Módulos sujetos a pruebas:

Este documento tiene como objetivo desarrollar un plan de pruebas para el Sistema SIAVEM, con el fin de verificar su correcto funcionamiento y asegurar su buen estado. Se definirán las diferentes pruebas necesarias para cada módulo y submódulo, incluyendo pruebas de funcionalidad, rendimiento y seguridad. El resultado final será un plan detallado que cumpla con los requisitos y expectativas de la Municipalidad de Santa Cruz.

- **Módulo Mantenimiento:**

1. Preventivo
2. Correctivo
3. Inspección

- **Módulo Unidad de transporte:**

1. Solicitud de gira

- **Módulo Usuarios:**

1. Registro de usuarios por departamentos

- **Módulo Vehículos:**

1. Ingreso de unidades vehiculares

- **Módulo Seguridad:**

1. Roles
2. Permisos

# Características a probar

**Mantenimiento Preventivo:** En este submódulo se realizará el listado de cada unidad móvil de transporte cada vez que se deba enviar al taller autorizado. Este mismo recibirá los datos de *fecha, tipo de mantenimiento, horimetro actual, kilometraje, Monto económico, contratación, taller, observaciones*.

**Mantenimiento Correctivo:** En este submódulo se realizará el listado de cada unidad móvil de transporte cada vez que este se encuentre en el taller recibiendo su debido mantenimiento. Este mismo recibirá los datos de *fecha de salida, fecha de regreso, horimetro actual, kilometraje, detalle, monto económico, contratación, taller, observaciones*.

**Mantenimiento Inspección:** En este submódulo se realizará la inspección de cada unidad solicitada una vez esta misma sea aprobada por el departamento de transporte. Este mismo recibirá los datos de *Fecha, Kilometraje, Placa, Combustible, Luces, Accesorios, Llantas, Espejos, Carrocerías*

**Solicitud de gira:** Será el objetivo principal del proyecto, en esta misma se realizará la solicitud para el uso de alguna unidad vehicular de la municipalidad de Santa Cruz, solo los jefes de cada departamento de la municipalidad serán los encargados de realizarlas, así mismo solamente los del área de transporte solamente serán los encargados de autorizar o denegar la solicitud. Se registrará *la fecha, hora de salida, hora de regreso, chofer, destino, objetivo, acompañantes*. Adicionalmente se adjuntan documentos de personas externas de la municipalidad que lleguen a utilizar alguna de las unidades municipales.

**Ingreso de unidades:** Este módulo será únicamente para el registro de todas las unidades que están a cargo del departamento de transporte. Este almacenará *Categoría, Tracción, combustible, estado, placa, marca, modelo, estilo, color, año de fabricación, valor de adquisición, valor hacienda, RTV, descripción*. Adicionalmente este módulo contará con la opción de editar, leer y eliminar unidades.

**Registro de usuarios por departamentos:** Este módulo está disponible únicamente para el registro de todos los empleados de cada departamento de la municipalidad de Santa Cruz, el

sistema almacenará el *departamento, puesto, nombre, apellidos, fecha de nacimiento, cédula, teléfono, correo institucional*. Adicionalmente este módulo contará con la opción de editar, leer y eliminar unidades.

**Roles:** Esta será un área a la que solo podrán acceder los administradores. Finalmente, se crearán usuarios con nombres de usuario y roles que contendrán las acciones que pueden realizar en el sistema. le permite crear, editar, borrar y ver.

**Permisos:** Esta también será área de administrador, desde la cual se otorgarán todos los permisos. Puede realizar acciones como guardar, ver, editar y eliminar.

## Definición de pruebas.

**Pruebas Unitarias:** Las pruebas unitarias son una técnica de prueba de software que verifica el correcto funcionamiento de una sección del código mediante pequeñas pruebas que validan el comportamiento y la lógica del objeto. Estas pruebas son esenciales para detectar errores en una fase temprana del desarrollo y garantizar la calidad y la funcionalidad del software.

Técnicamente, eso sería una clase o incluso un método de clase en lenguajes orientados a objetos, y un procedimiento o función en lenguajes procedimentales y funcionales.

Participantes: Ariana Leal Villafuerte, Abigaíl Espinoza Herrera

Metodología: Para desarrollar pruebas unitarias efectivas, es recomendable seguir la metodología de las "3A del unit testing", que consiste en un enfoque funcional que se divide en tres pasos:

1. Arrange(organizar): Primer paso es definir los requisitos necesarios para que el código cumpla con su objetivo.
2. Act(actuar): Segundo paso será ejecutar el código para visualizar sus resultados.



3. Assert(afirmar): Tercer y último paso será comprobar si la ejecución muestra resultados positivos, si no entonces se debe corregir el error hasta hacerlo funcionar.

## Pruebas de rendimiento y estrés

**Definición:** Las pruebas de rendimiento son fundamentales para asegurar un alto nivel de rendimiento del sistema. Estas pruebas permiten verificar si el sistema cumple con los criterios de rendimiento y validar atributos de calidad. También son útiles para identificar áreas problemáticas que afectan el rendimiento del sistema. Las pruebas de carga y estrés son dos tipos comunes de pruebas de rendimiento que se utilizarán en este proyecto.

**Pruebas de Carga:** Este es un tipo de prueba planificada que tiene como objetivo generar un número determinado de solicitudes en el sistema para comprobar su funcionamiento en situaciones de alta demanda. Mediante esta prueba, podemos evaluar los tiempos de respuesta de las transacciones y asegurarnos de que el sistema responda adecuadamente bajo carga.

**Pruebas de Estrés:** Esta prueba consiste en verificar cuántos usuarios puede soportar la plataforma, ya que se va doblando el número de usuarios, los cuales ejecutarán pruebas de carga hasta que la plataforma deje de funcionar.

### **Participantes:**

**Metodología:** Para llevar a cabo las pruebas de rendimiento de manera efectiva, se requieren estrategias bien planificadas que aseguren resultados adecuados. En primer lugar, es importante realizar una planificación detallada de estas pruebas para definir las razones por las que se aplicarán en el sistema y los objetivos que se esperan alcanzar. También es importante tener en cuenta que no todas las partes del sistema requieren pruebas de rendimiento.

Una vez definidos los objetivos, se debe identificar el entorno de pruebas para su evaluación. Esto implica identificar tanto el entorno físico de pruebas como el entorno de

producción, y tener en cuenta los recursos y herramientas disponibles para el equipo de pruebas. El entorno físico incluye hardware, software y configuraciones de red.

La siguiente etapa es identificar los criterios de aceptación de rendimiento para su selección. Se deben determinar el tiempo de respuesta, el rendimiento, la utilización de los recursos y los objetivos y limitaciones. En general, el tiempo de respuesta concierne al usuario, el rendimiento al negocio y la utilización de los recursos al sistema. También se deben identificar los criterios de éxito de rendimiento del proyecto para evaluar qué combinación de la configuración da lugar a un funcionamiento óptimo.

Una vez definidos los criterios de aceptación de rendimiento, se procede a planificar y diseñar las pruebas. Esto implica identificar los principales escenarios, determinar la variabilidad de los usuarios y cómo simular esa variabilidad, definir los datos de las pruebas y establecer las métricas que se recogerán.

Otra actividad importante es la configuración del entorno de prueba. Esto implica preparar el entorno, las herramientas y los recursos necesarios para ejecutar cada una de las estrategias de prueba, así como las características y componentes disponibles para la prueba.

La siguiente actividad es aplicar el diseño de la prueba. Esto implica desarrollar las pruebas de rendimiento de acuerdo con el diseño del plan previamente establecido.

Una vez que se han diseñado las pruebas, se procede a su ejecución y análisis de los resultados. Es importante consolidar y compartir los resultados de la prueba y analizar los datos, tanto individualmente como con un equipo multidisciplinario. Además, se debe preparar un informe detallado de los resultados obtenidos.

Para llevar a cabo estas pruebas, existen numerosas herramientas, como NeoLoad, LoadRunner, LoadUI, WebLOAD, entre otras. Una de las herramientas más populares es JMeter, que es una herramienta de código abierto.

# Pruebas de aceptación del usuario

**Definición:** La prueba de aceptación tiene como objetivo confirmar que el aplicativo web diseñado cumpla con las funciones y requisitos solicitados por la empresa patrocinadora.

Durante esta prueba, los usuarios finales del sistema (clientes) comparan el sistema con sus requisitos iniciales y verifican que cumpla con sus expectativas. Es una prueba crucial para garantizar la calidad del sistema.

## Tipos de prueba

### Pruebas de aceptación del usuario UAT

En realidad es una de las partes más importantes del proyecto ya que esta fase es dónde cabe la perspectiva de los usuarios finales y del conocimiento del dominio de un departamento que representa a los usuarios finales.

### Pruebas de aceptación operativa OAT

Se utilizan para realizar la preparación operativa o llamado pre lanzamiento de un producto, servicio o sistema como parte de un sistema de gestión de la calidad, esto es realizar una prueba de manera que se pueda minimizar el grado de error al poner el sistema en producción.

### Pruebas de aceptación de la normativa

Estas pruebas se realizan para garantizar que el sistema cumpla con los requisitos legales y de regulación aplicables, como la protección de datos personales, la privacidad, la seguridad, la accesibilidad, la interoperabilidad, entre otros.

**Nota importante:**

Es de suma importancia destacar que las pruebas de aceptación de la normativa en un sistema de software son esenciales para garantizar la legalidad y la ética del uso de los sistemas de software, y para evitar posibles sanciones legales y pérdidas financieras para la empresa o organización patrocinadora

## Pruebas de aceptación Alfa Y Beta

**Prueba Alfa:** Esta es una forma de prueba de aceptación interna realizada principalmente por los equipos internos de control de calidad y pruebas del software. La prueba alfa es la última prueba realizada por los equipos de prueba en el sitio de desarrollo después de la prueba de aceptación y antes de lanzar el software para la prueba beta.

**Prueba Beta:** Esta es una etapa de prueba sumamente importante seguida por el ciclo de prueba alfa completo interno.

**Nota Importante:** Esta es la fase de prueba final en la que las empresas lanzan el software a algunos grupos de usuarios externos fuera de los equipos de prueba o empleados de la empresa. Esta versión inicial del software se conoce como versión beta. La mayoría de las empresas recopilan comentarios de los usuarios en esta versión.

### Participantes:

Yustin López Salas, Daniela Chaves Aguirre

### Metodología:

**Metodología:** Los pasos para realizar una prueba de aceptación como actividad independiente para dar seguridad en la calidad del software desarrollado.

Es importante destacar cada uno de los siguientes puntos definidos.

**Definir los criterios de aceptación:** Es necesario establecer los criterios de aceptación que el software debe cumplir para ser considerado exitoso. Esto puede incluir los requisitos de funcionalidad, desempeño, seguridad, facilidad de uso y otros aspectos relevantes para el software.

**Diseñar la prueba:** En este paso, es necesario diseñar los casos de prueba que se utilizarán para verificar que el software cumpla con los criterios de aceptación definidos. Los casos de prueba deben ser exhaustivos, cubriendo todas las funcionalidades y escenarios de uso esperados del software.

**Preparar el entorno de prueba:** Es necesario preparar el entorno de prueba para que sea similar al entorno de producción. Esto puede incluir la instalación del software, la configuración de la red y la preparación de los datos de prueba.

**Ejecutar la prueba:** Una vez que se han diseñado los casos de prueba y se ha preparado el entorno de prueba, se pueden ejecutar las pruebas. Es importante registrar los resultados de las pruebas y documentar cualquier problema o defecto que se encuentre.

**Informar los resultados:** Una vez finalizada la prueba, es necesario informar los resultados a los interesados, incluyendo los desarrolladores, los gerentes de proyecto y los usuarios finales. Es necesario presentar los resultados de una manera clara y concisa para facilitar la comprensión y la toma de decisiones.

**Retestear los problemas encontrados:** Si se encuentran problemas durante la prueba, es necesario que los desarrolladores corrijan los problemas identificados y vuelvan a realizar las pruebas para asegurarse de que se han solucionado.

**Aceptar o rechazar el software:** Una vez que se han corregido todos los problemas identificados y se han realizado las pruebas necesarias, se debe tomar la decisión final de aceptar o rechazar el software.

- Nota importante: Esto se basará en los criterios de aceptación previamente definidos y los resultados de la prueba de aceptación. Si el software cumple con todos los criterios de aceptación, se puede aceptar y poner en producción. De lo contrario, se deben tomar medidas para corregir los problemas antes de volver a realizar la prueba de aceptación.

## Principales entregables

La documentación de los entregables durante la aplicación de diferentes tipos de pruebas de software es fundamental, ya que permite definir los tipos de prueba a aplicar, la metodología a utilizar y registrar los resultados obtenidos. A medida que se llevan a cabo las pruebas, se generan diversos documentos que ayudan a mantener un registro detallado de los procedimientos y resultados de las pruebas. Es importante destacar que la documentación de los entregables contribuye a la gestión eficiente de las pruebas y a mejorar la calidad del software, esta opinión la basamos en lo antes investigado.

- Plan de prueba
- Casos de prueba
- Probar informes de incidentes
- Informes de resumen de pruebas.

### Riesgos

1- Las pruebas realizadas en el aplicativo web evidencian que no se cumplen con los requerimientos solicitados.

2- Si por ejemplo la entrega de los elementos de prueba se retrasa, es posible que sea necesario programar más turnos de noche o madrugada para poder dar con el cumplimiento de la fecha de entrega.

### **Nota Importante:**

Además de los riesgos mencionados, existen otros que se pueden experimentar durante el proceso de pruebas de software, tales como:

**Errores en los datos de prueba:** Si los datos de prueba no son adecuados o están incompletos, puede haber resultados inesperados en las pruebas que podrían llevar a conclusiones erróneas sobre la calidad del software.

**Falta de recursos:** Si no se cuenta con suficientes recursos, como personal capacitado o herramientas adecuadas, puede ser difícil realizar pruebas efectivas.

**Cambios en los requerimientos:** Si se realizan cambios en los requerimientos durante el proceso de pruebas, puede haber que se deba ajustar la estrategia y los planes de pruebas, lo que podría afectar la calidad del software.

## **Herramientas**

**Lighthouse:** es una herramienta gratuita y de código abierto de Google que permite realizar pruebas automáticas en aplicaciones web y determinar si cumplen con los requisitos para ser consideradas Aplicaciones Web Progresivas (PWA).

**PHPUnit:** Es una herramienta de prueba unitaria que se utiliza en el entorno para el lenguaje de programación PHP.

**Google form:** Herramienta importante para los formularios creados con Google Forms porque pueden ser utilizados para recopilar información de manera eficiente, ya sea para

realizar encuestas de opinión, recopilar datos. Además, Google Forms es una herramienta muy versátil que permite personalizar los formularios con diferentes tipos de preguntas

## Bibliografía

*Ejemplo de documento de plan de prueba (ejemplo de plan de prueba con detalles de cada campo).* (s/f). Myservname.com. Recuperado el 26 de marzo de 2023, de

<https://spa.myservname.com/sample-test-plan-document>

NimbleWork, P. R. (2022, diciembre 19). *Pruebas De Aceptación: El Qué Y Porqué + Los Tipos Que Hay Que Conocer*. Nimblework; NimbleWork, Inc.

<https://www.nimblework.com/es/agile/pruebas-de-aceptacion/>

Avalos, S. S. (2019, septiembre 26). *Pruebas de carga vs Pruebas de estrés*. Somos PNT - Desarrollamos Software. <https://sospnt.com/blog/103-load-testing-vs-stress-testing>

NimbleWork, P. R. (2022b, diciembre 22). *Prueba Unitaria: Un Tutorial Sobre Qué Es, Cómo Hacerlo + Herramientas Para Usar*. Nimblework; NimbleWork, Inc.

<https://www.nimblework.com/es/agile/pruebas-unitarias/>

Yeeply. (2022, septiembre 6). *¿Qué son las pruebas unitarias y cómo llevar una a cabo?*

Yeeply. <https://www.yeeply.com/blog/que-son-pruebas-unitarias/>

¿Qué es la prueba de aceptación del usuario (UAT): una guía completa? - Tipos De Pruebas. (s.f.). Los Comentarios, Juegos, Entretenimiento, Marzo 2023.

[https://spa.myservname.com/what-is-user-acceptance-testing#When\\_Is\\_It\\_Performed](https://spa.myservname.com/what-is-user-acceptance-testing#When_Is_It_Performed)



¿Qué es la prueba de aceptación del usuario (UAT): una guía completa? - Tipos De Pruebas. (s.f.). Los Comentarios, Juegos, Entretenimiento, Marzo 2023.

[https://spa.myservername.com/what-is-user-acceptance-testing#When\\_Is\\_It\\_Performed](https://spa.myservername.com/what-is-user-acceptance-testing#When_Is_It_Performed)

¿Qué son las pruebas alfa y las pruebas beta? Una guía completa - Tipos De Pruebas. (s.f.). Los Comentarios, Juegos, Entretenimiento, Marzo 2023.

<https://spa.myservername.com/alpha-testing-beta-testing>

